

Sidebar

Sidebar This document outlines the specifications for the Sidebar Component, a React component designed to provide a customizable sidebar for web applications.

Behavior Description ■ The Sidebar Component should include a logo at the top and a clickable icon/button positioned towards the top right for expanding and collapsing the sidebar. When collapsing the sidebar, only the icons of sidebar items should be displayed. Each item within the sidebar should consist of an icon and a text label with appropriate padding. Clicking on any item should activate it, highlighting it to indicate its active state.

Additionally, each **SidebarItem** Component may contain children. In these instances, the parent **SidebarItem** should feature a dropdown icon at its far right. Clicking on a **SidebarItem** with children should expand the sidebar, revealing its child items. Collapsing the sidebar should only reveal the icons of sidebar items.

Inputs ■ **Props** ■ **logo** : The logo to display at the top of the sidebar. **children** : The Sidebar component should accept **SidebarItem** components as its children. **onItemClick** : A callback function triggered when an item in the sidebar is clicked. It should accept a parameter to receive the clicked item's data. **onCollapseToggle** : A callback function triggered when the collapse/expand button is clicked. **ToggleIcon** : The logo to display at the top left of the sidebar, serving as a toggle for collapsing and expanding the sidebar.

Details on Inputs ■ **logo** Type: `ReactNode` (e.g., `JSX`, `HTML` element) Required: Yes Description: The logo to display at the top of the sidebar. **children** (**SidebarItem**) Type: `ReactNode` Required: Yes Description: The Sidebar component accepts **SidebarItem** components as its children. **onItemClick** Type: `Function` Required: Yes Parameters: `label` (`string`): The label of the clicked sidebar item.

Description: A callback function that receives the label of the clicked sidebar item when an item in the sidebar is clicked. **onCollapseToggle** Type: `Function` Required: Yes

Description: A callback function triggered when the collapse/expand button is clicked.

ToggleIcon Type: `ReactNode` (e.g., `JSX`, `HTML` element) Required: Yes Description: The logo to display at the top left of the sidebar. It serves as a toggle for collapsing and expanding the sidebar. When clicked, it triggers the **onCollapseToggle** callback function.

SidebarItem Component ■ **Props** ■ **Icon** : Icon to display when Sidebar is collapsed. **label** : The text to display for the sidebar item. **link** : The URL or route path to navigate to when the item is clicked. **active** : Boolean indicating whether the item is currently active or not. **children** (Optional): Child **SidebarItems** for creating nested items. < Sidebar logo = { < img

src = " logo.png " alt = " Logo " /> } onItemClick = { handleItemClick } onCollapseToggle = { handleCollapseToggle } > < SidebarItem label = " Home " link = " / " active = { true } /> < SidebarItem label = " Profile " link = " /profile " > < SidebarItem label = " Settings " link = " /profile/settings " /> </ SidebarItem > < SidebarItem label = " Settings " link = " /settings " /> </ Sidebar > Conclusion ■ With the provided documentation, you should now be able to create and integrate the Sidebar component along with SidebarItem child components into your TypeScript-based React applications effectively.

```
<Sidebar
  logo={}
  onItemClick={handleItemClick}
  onCollapseToggle={handleCollapseToggle}
>
  <SidebarItem label="Home" link="/" active={true} />
  <SidebarItem label="Profile" link="/profile">
    <SidebarItem label="Settings" link="/profile/settings" />
  </SidebarItem>
  <SidebarItem label="Settings" link="/settings" />
</Sidebar>
```